UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/585,292	07/05/2006	Rongzhen Yang	P20651	2366

59796   7590   09/13/2010
INTEL CORPORATION
c/o CPA Global
P.O. BOX 52050
MINNEAPOLIS, MN 55402

EXAMINER
MITCHELL, JASON D

ART UNIT	PAPER NUMBER
2193	

MAIL DATE	DELIVERY MODE
09/13/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

	Application No.	Applicant(s)
Office Action Summary	10/585,292	YANG ET AL.
	Examiner	**Art Unit**	
	Jason Mitchell	2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on <u>05 July 2006</u>.
2a) ☐ This action is **FINAL**.   2b) ☒ This action is non-final.
3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) <u>1-20</u> is/are pending in the application.
    4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5) ☐ Claim(s) _____ is/are allowed.
6) ☒ Claim(s) <u>1-20</u> is/are rejected.
7) ☐ Claim(s) _____ is/are objected to.
8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) ☐ The specification is objected to by the Examiner.
10) ☒ The drawing(s) filed on <u>05 July 2006</u> is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
    a) ☒ All   b) ☐ Some * c) ☐ None of:
      1. ☒ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____.
      3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date <u>4/7/08</u>.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

U.S. Patent and Trademark Office
PTOL-326 (Rev. 08-06)   Office Action Summary   Part of Paper No./Mail Date 20100825

DETAILED ACTION

This action is in response to an application filed on 7/5/06.

Claims 1-20 are pending in this application.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 9-16 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claim 9 is not limited to statutory embodiments. In view of Applicant's disclosure, specification page 15, par. [0051], the claimed medium is not limited to statutory embodiments, instead being defined as including both statutory embodiments (e.g., "storage media such as floppy disks ...") and non-statutory embodiments (e.g., communications media such [as] electromagnetic or optical carriers"). As such, the claim is not limited to statutory subject matter and is therefore non-statutory.

Claims 10-16 depend from claim 9 and are rejected accordingly.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claims 8 and 16 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 8 recites "collecting performance information corresponding to instruction addresses for substantially all instances of the common code segment" The term "substantially" is a relative term which renders the claim indefinite. The term "substantially" is not defined by the claim, the specification does not provide a standard for ascertaining the requisite degree, and one of ordinary skill in the art would not be reasonably apprised of the scope of the invention.

Claim 16 recites a limitation similar to claim 8 and is rejected accordingly.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1, 3-4, 7-9, 11-12, 15-17 and 19-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 5,212,794 to Pettis et al. (Pettis) in view of "A Language Independent Approach for Detecting Duplicated Code" by Ducasse et al. (Ducasse).

Claims 1, 9 and 17: Pettis discloses a method comprising:

obtaining performance data for software that has executed in a data processing

system, wherein the performance data comprises instruction addresses and

corresponding performance information (col. 2, lines 51-54 "statistical information

obtained by running the computer code"; the disclosed information must include

information for mapping the executed instruction(s) to corresponding instructions in the

source code, this at least broadly constitutes an address);

obtaining dump information from the data processing system, wherein the dump

information comprises the instructions and corresponding instruction addresses (col. 2,

lines 59-61 "The first computer program comprising a set of basic blocks of computer

code arranged in a first linear order"; the disclosure of a 'linear order' indicates that this

data includes at least local or relative address information);

automatically identifying common code segments in the performance data,

wherein a common code segment comprises an ordered set of multiple instructions that

appears multiple times in the performance data (col. 9, lines 38-47 "The linkage

placement of the procedures is determined by constructing groups similar to the chains

described above with reference to basic blocks … by first choosing the edge with the

heaviest weight"; note that a procedure constitutes a set of ordered instructions and the

disclosed grouping of procedures constitutes an identification of a superset of ordered

instructions which appear frequently in the trace); and

generating aggregate performance data for the common code segments, based

at least in part on the instruction addresses associated with the common code

segments from the dump information, the instruction addresses from the performance

data, and the corresponding performance information from the performance data (col. 9,

lines 50-53 "The two nodes are merged into a single node or group having two

procedures therein"; col. 9, lines 30-34 "If a procedure calls another from several

different places within itself, ... those weights are first summed together to form the edge

weight shown in the graph").

Pettis does not disclose automatically identifying common code segments in the dump

information ... and generating aggregate performance data for the common code

segments in the dump information.

Ducasse teaches identifying common code segments in dump information (Abstract

"detect a significant amount of code duplication").

It would have been obvious to one of ordinary skill in the art at the time the invention

was made to use Ducasse's techniques (Abstract "detect a significant amount of code

duplication") to identify duplicate code segments in Pettis' dump information (col. 2,

lines 59-61 "computer program comprising a set of basic blocks") and to aggregate data

for these duplicated code sections (see e.g. Pettis "weights are first summed together to

form the [aggregated] edge weight"). Those of ordinary skill in the art would have been

motivated to do so because "[c]ode duplication increases the size of the code ...

expanding the size of the executable" (see Ducasse pg. 1, col. 2, 1st partial par.) and

Pettis attempts to reduce the size of 'frequently executed' code loaded in memory (see

e.g. Pettis col. 7, lines 44-50 "minimize the size of the primary part of the procedures").

In other words, unrecognized code duplication could result in Pettis' system two or more

separate chains (e.g. Clone1—P and Clone2—P) which would more optimally be

represented as a single chain (e.g. Clone1/Clone2—P). This single chain representation

would result in a smaller "primary part" of the program being loaded into memory and a

more accurate weight being applied to the behavior so that the importance of this chain

can more accurately be recognized. Further note that Ducasse explicitly suggest such a

combination (pg. 9, col. 2, 1st partial par. "Investigate if textual comparison is a useful

approach for exploring the structure of ... program execution traces, wherein it could be

interesting to find recurring patterns of execution sequences")

Claims 3, 11 and 19: The rejections of claims 1, 9 and 17 are incorporated; further

Ducasse teaches the operation of identifying common code segments in the dump

information comprises:

    selecting a candidate code segment from the dump information (pg. 3, col. 1, 1st

par. "compare every entity ... with every other entity");

    determining whether the candidate code segment occurs multiple times in the

dump information (pg. 3, col. 1, 1st par. "compare every entity ... with every other

entity"); and

    identifying the candidate code segment as a common code segment in response

to determining that the candidate code segment occurs multiple times in the dump

information (pg. 3, col. 1, 1st par. "The result is a Boolean true for an exact match").

Claims 4 and 12: The rejections of claims 1, 9 are incorporated; further Ducasse

teaches the operation of identifying common code segments in the dump information

comprises:

selecting a candidate code segment from the dump information (pg. 3, col. 1, 1st

par. "compare every entity ... with every other entity");

determining whether the dump information includes at least one additional

absolute match for the candidate code segment (pg. 3, col. 1, 1st par. "compare every

entity ... with every other entity"); and

identifying the candidate code segment as a common code segment in response

to determining that the dump information includes at least one additional absolute match

for the candidate code segment (pg. 3, col. 1, 1st par. "The result is a Boolean true for

an exact match").

Claims 7, 15 and 20: The rejections of claims 1, 9 and 17 are incorporated; further

Pettis and Ducasse teach:

the operation of identifying common code segments in the dump information

comprises identifying at least first and second common code segments (Ducasse pg. 3,

col. 1, 1st par. "compare every entity ... with every other entity"); and

the operation of generating aggregate performance data for the common code

segments comprises:

collecting performance data for multiple instances of the first common

code segment (Pettis col. 2, lines 51-54 "statistical information obtained by

running the computer code");

generating aggregate performance data for the first common code

segment, based at least in part on the performance data for the multiple

instances of the first common code segment (Pettis col. 9, lines 50-53 "The two

nodes are merged into a single node");

collecting performance data for multiple instances of the second common

code segment (Pettis col. 2, lines 51-54 "statistical information obtained by

running the computer code"); and

generating aggregate performance data for the second common code

segment, based at least in part on the performance data for the multiple

instances of the second common code segment (Pettis col. 9, lines 50-53 "The

two nodes are merged into a single node").

Claims 8 and 16: The rejections of claims 7, 15 are incorporated; further Pettis

discloses the operation of generating aggregate performance data for the common code

segments comprises:

collecting performance information corresponding to instruction addresses for

substantially all instances of the common code segment in the dump information (col. 2,

lines 51-54 "statistical information obtained by running the computer code").

Claims 2, 6, 10, 14 and 18 are rejected under 35 U.S.C. 103(a) as being

unpatentable over US 5,212,794 to Pettis et al. (Pettis) in view of "A Language

Independent Approach for Detecting Duplicated Code" by Ducasse et al.

(Ducasse) in view of .

Claims 2, 10 and 18: The rejections of claims 1, 9 and 17 are incorporated; further

Pettis and Ducasse do not teach: obtaining performance data for instructions generated

by a dynamic compiler; and generating aggregate performance data for common code

segment generated by the dynamic compiler.

Chauvel teaches obtaining performance data for instructions generated by a dynamic

compiler; and generating aggregate performance data for common code segment

generated by the dynamic compiler (par. [0012] "acquiring a virtual machine profile that

relates a performance characteristic to individual operations and generating an

aggregate value for the performance characteristic based on the application profile and

the virtual machine profile").

It would have been obvious to one of ordinary skill in the art at the time the invention

was made obtain and aggregate performance data as taught by Pettis and Ducasse

(Pettis col. 2, lines 51-54 "statistical information obtained by running the computer

code"; col. 9, lines 50-53 "The two nodes are merged into a single node or group having

two procedures therein") for instructions generated by a dynamic compiler as taught by

Chauvel (par. [0012] "acquiring a virtual machine profile"). Those of ordinary skill in the art would have been motivated to do so in order to optimize an application compiled by a dynamic compiler (par. [0011] "a need has arisen for a method and apparatus for reliably estimating performance characteristics, ... in a device using a virtual machine interface").

Claims 6 and 14: The rejections of claims 1, 9 and are incorporated; further Pettis and Ducasse do not explicitly teach the performance information comprises one or more measurements selected from the group consisting of: execution time data for individual instructions; and cache miss data for individual instructions.

Chauvel teaches performance information comprises one or more measurements selected from the group consisting of: execution time data and cache miss data (par. [0025] "performance characteristics such as … cache misses … operation execution time").

It would have been obvious to one of ordinary skill in the art at the time the invention was made to gather performance information comprising execution time data or cache miss data as taught by Chauvel (par. [0025] "performance characteristics such as … cache misses … operation execution time") in the performance information gathered Pettis (Pettis col. 2, lines 51-54 "statistical information obtained by running the computer code"). Those of ordinary skill in the art would have been motivated to do so would

allow the tool to focus on the most problematic areas of the program (see e.g. col. 4,

lines 30-32 "there are fewer cache misses"; Chauvel par. [0025] "cache misses").

Claims 5 and 13 are rejected under 35 U.S.C. 103(a) as being unpatentable over

US 5,212,794 to Pettis et al. (Pettis) in view of "A Language Independent

Approach for Detecting Duplicated Code" by Ducasse et al. (Ducasse) in view of

"A Program for Identifying Duplicated Code" by Baker (Baker).

Claims 5 and 13: The rejections of claims 1, 9 and are incorporated; further Ducasse

teaches the operation of identifying common code segments in the dump information

comprises:

selecting a candidate code segment from the dump information (pg. 3, col. 1, 1st

par. "compare every entity … with every other entity");

determining whether the dump information includes at least one additional match

for the candidate code segment (pg. 3, col. 1, 1st par. "compare every entity … with

every other entity"); and

identifying the candidate code segment as a common code segment in response

to determining that the dump information includes at least one additional match for the

candidate code segment (pg. 3, col. 1, 1st par. "The result is a Boolean true for an exact

match").

Pettis and Ducasse do not teach identifying elements in the candidate code segment as significant; and wherein the additional match comprises instructions with elements matching the significant elements in the candidate code segment.

Baker teaches identifying elements in the candidate code segment as significant; and determining whether the dump information includes at least one additional match for the candidate code segment, wherein the additional match comprises instructions with elements matching the significant elements in the candidate code segment (pg. 2, par. bridging cols. 1 & 2 "the program can look for parameterized matches, where the code sections match except for a one-to-one correspondence between candidates for parameters such as variables").

It would have been obvious to one of ordinary skill in the art at the time the invention was made to substitute Baker's "parameterized" matching algorithm (pg. 2, par. bridging cols. 1 & 2 "the program can look for parameterized matches") for Decease's "absolute" matching algorithm (pg. 3, col. 1, 1st par. "The result is a Boolean true for an exact match"). Those of ordinary skill in the art would have been motivated to do so because parameterized matching will result in the identification of more duplication (pg. 3, col. 1, 1st full par. "code will have more parameterized matches than exact matches") and thus greater optimization (as discussed in the rejection of the parent claim(s)).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jason Mitchell whose telephone number is (571)272-3728. The examiner can normally be reached on Monday-Thursday and alternate Fridays 7:30-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Bullock Lewis can be reached on (571) 272-3759. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Jason Mitchell/
Primary Examiner, Art Unit 2193